Available online at www.sciencedirect.com
SCIENCE DIRECT®
Reliability Engineering and System Safety 86 (2004) 297–305
RELIABILITY ENGINEERING & SYSTEM SAFETY
www.elsevier.com/locate/ress

A fully Bayesian approach for combining multilevel failure information in fault tree quantification and optimal follow-on resource allocation

M. Hamada[a,*], H.F. Martz[a], C.S. Reese[b], T. Graves[a], V. Johnson[c], A.G. Wilson[a]

[a]*Group D-1, MS F600, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*
[b]*Department of Statistics, 230 TMCB, Brigham Young University, Provo, UT 84602, USA*
[c]*School of Public Health, Building II, 1420 Washington Heights, University of Michigan, Ann Arbor, MI 48105, USA*

Received 11 October 2003; accepted 10 February 2004

Abstract

This paper presents a fully Bayesian approach that simultaneously combines non-overlapping (in time) basic event and higher-level event failure data in fault tree quantification. Such higher-level data often correspond to train, subsystem or system failure events. The fully Bayesian approach also automatically propagates the highest-level data to lower levels in the fault tree. A simple example illustrates our approach. The optimal allocation of resources for collecting additional data from a choice of different level events is also presented. The optimization is achieved using a genetic algorithm.
Published by Elsevier Ltd.

Keywords: Genetic algorithm; Information gain; Markov chain Monte Carlo

1. Introduction

Vesely et al. [1], the Probabilistic Risk Assessment (PRA) Procedures Guide [2], and many other textbooks discuss fault tree quantification (e.g. the estimation of basic and higher-level event probabilities in a fault tree). This quantification consists of three steps: (1) determining the basic event probabilities, (2) calculating the minimal cut set probabilities, and (3) determining the system (i.e. the top event) probability using either exact or approximate methods.

It is current and accepted practice in fault tree and accident sequence quantification (as implemented, for example, in the Systems Analysis Programs for Hands-on Integrated Reliability Evaluations [SAPHIRE [3] package and the Integrated Reliability and Risk Analysis System [IRRAS [4,5]]]) to use only statistical data and information regarding the basic events. In a departure from this practice, Martz and Almond [6] use non-overlapping statistical data and information collected on higher-level events or gates in the tree to modify standard estimates. Doing so is important because normal operation and testing procedures often generate data for many high-level gates corresponding to, for example, train, subsystem, and system unavailability, and often even the top event itself.

By 'non-overlapping' we mean that the higher-level event data are from either non-overlapping time periods or demands. Otherwise, the use of higher-level event data would result in double counting of data and thus dependency. This 'non-overlapping' constraint naturally applies to any system test that is destructive, such as a missile fired at a target. If the same higher-level data provide basic event information, then we can instead use such data to verify the structure of the fault tree. In particular, any higher-level failure data not consistent with the fault tree is an indication that the fault tree model is inadequate. Note that data from overlapping subsystems, i.e. which consist of some of the same basic events, can also be incorporated as long as they are non-overlapping in time; data overlapping in time can also be employed if the subsystems are independent such as data collected from different plants.

This paper describes a fully Bayesian approach which can simultaneously combine basic event and independent higher-level failure data and information in fault tree quantification. The obvious advantage of this approach is the associated increase in accuracy and precision of estimated probabilities that result from the combined use

* Corresponding author.
E-mail address: hamada@lanl.gov (M. Hamada).

0951-8320/$ - see front matter Published by Elsevier Ltd.
doi:10.1016/j.ress.2004.02.001

of these data. Note that Martz and Almond [6] only approximated the fully Bayesian approach by using method-of-moment type estimators based on the first two distributional moments.

The fully Bayesian approach also permits the incorporation of independent industry-wide statistical analyses that are sometimes performed on safety systems considered in a PRA. Such analyses represent a source of generic higher-level statistical information for the specific plant under consideration. For example, Grant et al. [7] describe an industry-wide statistical analysis of the safety-related performance of the high-pressure coolant injection system at US commercial boiling water reactor plants for the period 1987–1993.

1.1. Related methods

Numerous articles discuss system reliability for systems described by reliability block diagrams in which both component and independent system-level test data are combined. Mastran [8] and Mastran and Singpurwalla [9] consider an approximate Bayesian approach to the estimation of system reliability based on pass/fail test data collected at both the component and system levels for a coherent system of nonidentical components. They use a top-down approach that apportions the posterior system reliability distribution to each component through a component prior distribution that is consistent with the system configuration. By combining these component priors with the component level data, component posterior distributions are obtained. Propagating these component posteriors back up to the system level using the system model yields the final system posterior.

Martz et al. [10] and Martz and Waller [11] develop an approximate Bayesian procedure for estimating system reliability based on a bottom-up approach. In their approach, prior means and variances of prior distributions are combined with data and propagated upward in the system to obtain a system reliability posterior distribution.

Johnson et al. [12] propose a fully Bayesian approach for system reliability estimation for systems described by a reliability block diagram. This approach resolves the upward and downward propagation problem by simultaneously modeling the complete set of system parameters. We generalize their procedure in this paper to fault tree quantification.

When the higher-level and basic event data are overlapping, the above methods cannot be applied because the models do not account for the resulting dependent data. For example, a standby system may fail to operate upon demand (a higher-level system failure), and this failure may subsequently be traced to the failure of a particular component in the system (a basic event failure). However, the above methods (and the method presented here as well) are still applicable if only one level of data is used. Using the higher-level event data to form an aggregated posterior for the higher-level gate produces an *aggregate* analysis. Using the data at the basic event to form a disaggregated posterior for the higher-level event produces a *disaggregate* analysis. Usually, the aggregate and disaggregate posteriors will disagree, in which case we say that an *aggregation error* occurs. Very large aggregation errors are often grounds for suspicion of the structure of the fault tree model.

The concept of aggregation error is quite well-known and has been widely studied. It had its genesis in econometrics in the work of Simon and Ando [13], Ijiiri [14], and Chipman [15]. The book by Theil [16] describes its early developmental ideas. Mosleh and Bier [17] were the first to discuss aggregation error in the context of risk and reliability analysis. Bier [18] and Azaiez and Bier [19] likewise consider aggregation error in the Bayesian estimation of reliability.

An outline of the paper is as follows. In Section 2, to focus attention, we present an example fault tree. A Bayesian approach for using independent higher-level failure data in any coherent fault tree is presented in Section 3 and the required numerical Bayesian computations are summarized in Section 4. Section 5 illustrates the performance of the proposed approach using the fault tree example. Section 6 discusses the problem of allocating additional resources to improve inference of the top event probability. Section 7 concludes with a discussion.

2. Example

Consider the following simple fault tree example as depicted in Fig. 1. This fault tree was previously analyzed by Russell et al. [3,4] to illustrate the IRRAS fault tree methodology. It consists of AND and OR gates and one 2/3 gate. There are five basic events denoted by BE1–BE5. One intermediate event denoted by IE is identified and the top event is denoted by TE. Note the difference between a fault tree and a reliability block diagram in which, for our example, a basic event such as BE1 shows up in more than one branch of the fault tree. In this paper, we consider the situation where prior information and/or data are available at the basic, intermediate and top events.

3. A fully Bayesian approach for inference

We assume that the prior information about the probability of occurrence of each basic event can be summarized by a Beta(a, b) distribution. If additional basic event data are available in the form of binomial data, x failures, say, in n trials, then the available information for the basic event by combining the prior information and data can be obtained by Bayes' theorem. If additional basic event data are available, then the available information for the basic event can be expressed as a Beta$(a + x, b + n - x)$ distribution.

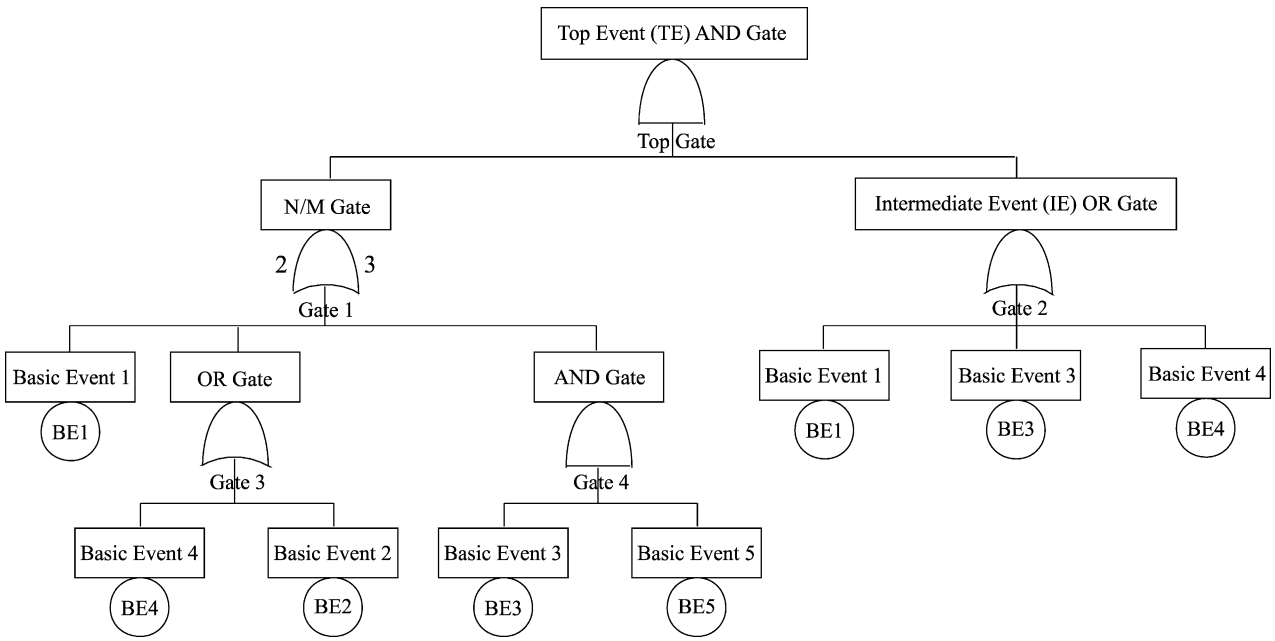

Fig. 1. Example fault Tree.

The proposed method also requires that the higher-level event information be expressed in terms of equivalent observational data; this requirement ensures that the posterior distribution of the basic event probability obtained using multilevel data and information is well defined. Thus, we express the higher-level event information as equivalent $\tilde{x}$ failures in $\tilde{n}$ trials, although $\tilde{x}$ and $\tilde{n}$ need not be integers. For example, suppose we believe that a higher-level event probability is 0.05, but that our belief is only as precise as information contained in two observations. In that case, we would set $\tilde{x} = 0.1$ and $\tilde{n} = 2$, i.e. 0.1 event occurrences in two trials. Note that this higher-level event information needs to be independent of that induced by propagating the basic event priors through the fault tree. In addition, if there are actual higher-level event data available, e.g. 3 events in 100 trials, then the combined information can be represented as 3.1 events in 102 trials or $\tilde{x} = 3.1$ and $\tilde{n} = 102$. If the higher-level information consists only of actual event data, then $\tilde{x}$ and $\tilde{n}$ are necessarily integers.

Following Johnson et al. [12], a key feature of the proposed method is that higher-level event probabilities are expressed in terms of basic event probabilities. For fault trees, these expressions can be obtained by determining the minimal cut sets of higher-level events and applying the law of total probability (also known as the inclusion–exclusion rule). For example, in the fault tree depicted in Fig. 1, the top event has five minimal cut sets:

$$\{\mathrm{BE1}, \mathrm{BE2}\},\ \{\mathrm{BE1}, \mathrm{BE4}\},\ \{\mathrm{BE1}, \mathrm{BE3}, \mathrm{BE5}\},$$

$$\{\mathrm{BE2}, \mathrm{BE3}, \mathrm{BE5}\},\ \{\mathrm{BE3}, \mathrm{BE4}, \mathrm{BE5}\}.$$

Using the law of total probability, the top event probability may then be expressed in terms of the basic events as

$$\begin{aligned}\mathrm{TE}(\mathbf{p}) = {} & p1p2 + p1p4 + p1p3p5 + p2p3p5 + p3p4p5 \\ & - p1p2p4 - 2p1p2p3p5 - 2p1p3p4p5 \\ & - p2p3p4p5 + 2p1p2p3p4p5, \end{aligned} \tag{1}$$

where $p1, \ldots, p5$ are the occurrence probabilities for basic events $\mathrm{BE1}, \ldots, \mathrm{BE5}$.

Similarly, the intermediate event probability can be expressed as

$$\begin{aligned}\mathrm{IE}(\mathbf{p}) = {} & p1 + p3 + p4 - p1p3 - p1p4 - p3p4 \\ & + p1p3p4. \end{aligned} \tag{2}$$

From these expressions of higher-level event probabilities, we see that higher-level event information provides information about basic event probabilities. Likewise, basic event information provides information about higher-level event probabilities.

4. Bayesian computation

As mentioned in Section 1, we obtain estimates of basic event probabilities through the Bayesian approach to inference. Bayesian methods are named for Bayes' theorem

$$\pi(\mathbf{p}|x) = \frac{f(x|p)\pi(p)}{\int f(x|p)\pi(p)\mathrm{d}p}, \tag{3}$$

where $\pi(\mathbf{p}|x)$ is called the posterior distribution, and is the conditional distribution of the unknown failure probability $\mathbf{p}$ given the observed data x. Furthermore, $f(x|p)$ is the sampling density (commonly referred to as the likelihood)

and $\pi(\mathbf{p})$ represents the prior distribution for $\mathbf{p}$. This prior distribution can be obtained from experts, computer models, engineering or physics theory, or previous studies. If there is no information about $\mathbf{p}$ before a study is conducted, a distribution which contains little or no information about $\mathbf{p}$ can be substituted, often referred to as a noninformative prior distribution. In our experience, there almost always exists some prior knowledge that can and should be incorporated.

Bayesian methods were relegated to obscurity for a long period of statistical history. The primary reason was that when $\mathbf{p}$ is of high dimension, the denominator in Eq. (3) was difficult (and sometimes impossible) to calculate. However, Gelfand and Smith [20] introduced computing routines that made computation of the denominator possible through simulation and Monte Carlo integration; Casella and George [21] and Chib and Greenberg [22] provide good introductions to these computing routines. The broad class of modern Bayesian computation was aptly named Markov chain Monte Carlo (MCMC) and Gilks et al. [23] provide a nice review of the basic elements of MCMC computation. At the heart of most basic Bayesian computation is the complete conditional (or full conditional) distribution which is defined as the conditional distribution of each parameter given all other parameters in the model, including the data. MCMC relies on the fact that sequential simulation from complete conditionals (replacing recently updated parameters successively) converges to the joint posterior distribution of all the parameters. So, given a starting point, after a certain number of preliminary iterations (called the burn-in period) the simulated observations will be from the desired joint posterior distribution. Often, simulation from a complete conditional is difficult (or seemingly impossible). The Metropolis-Hastings algorithm [22] is a method for simulating from an arbitrary distribution whose form is known up to a constant (as is the case with Bayesian computation). The central idea is that a random variable is generated from any distribution with probability density function $g(\cdot)$, and is accepted with probability

$$\min\left(1, \frac{g(z|y)h(y)}{g(y|z)h(z)}\right),$$

where z is the current value of the parameter (say, p) and y is the proposed replacement value of the parameter; here $h(\cdot)$ is probability density function (up to a constant) of the desired arbitrary distribution. As the algorithm proceeds, this distribution converges to the distribution of the actual complete conditional.

This is an amazing result and makes Bayesian computation available for a rich class of problems. One obvious consequence of the above choice is that the realizations will not be independent, but will almost certainly exhibit autocorrelation. In order to remedy this problem, it is often recommended that realizations be skipped and only every third observation, for example, be kept for inference. This process of dropping observations to approximate independence is called 'thinning'. Also, to remove dependence on the starting values of the parameters, a burn-in period is often employed which means that a number of the initial realizations are dropped before thinning subsequent realizations.

In our case inference is then obtained using Bayes' theorem implemented by MCMC; that is, we end up with a set of draws from the joint posterior distribution of the basic event probabilities $\mathbf{p}$.

The advantage of this fully Bayesian approach is that, except for the Monte Carlo sampling error which is controlled by taking more samples, no approximations are being made. The top event posterior distribution is based on all available data and the basic event posterior distributions are updated based on all higher-level data. We will apply the proposed procedure for the simple fault tree example under different scenarios in Section 5.

5. Example revisited

We consider several cases to examine the performance of the method as a function of two factors: the strength of the basic event data (strong or weak), the strength of the top event data (strong or weak). The results for each of these cases are compared and used as a means of assessing the performance of the proposed approach. Weak data correspond to a coefficient of variation (a ratio of the beta standard deviation to the mean) of approximately 2.5, while strong data have a coefficient of variation of approximately 0.4. The weak data roughly correspond to an equivalent lognormal error factor of 10, while the strong data roughly represent a lognormal error factor of 2. The lognormal error factors were also considered in Martz and Almond [6].

In the following tables, BE, TE and IE refer to basic, top and intermediate events, respectively. First, the cases considered can be classified according to whether information about the events is available and if so, whether it is weak or strong. Table 1 describes the cases in these terms.

Table 1
Various cases classified according to availability of event information

Case	BE1–BE5	TE	IE
1	Weak	Weak	None
2	Weak	Strong	None
3	Strong	Weak	None
4	Strong	Strong	None
5	Very weak	Strong	None
6	Strong	None	None
7	Very weak	Weak	None
8	Weak	None	None
9	Weak	Weak	Weak
10	Weak	Weak	Strong

Table 2
Various cases in terms of beta parameters (a, b) and equivalent data $(\tilde{x}, \tilde{n})$

Case	BE1[a]	BE2[a]	BE3[a]	BE4[a]	BE5[a]	TE[b]	IE[b]
1	0.152, 15.092	0.142, 6.899	0.129, 4.176	0.118, 2.821	0.106, 2.012	0.163, 162.923	
2	0.152, 15.092	0.142, 6.899	0.129, 4.176	0.118, 2.821	0.106, 2.012	5.141, 5140.881	
3	5.086, 503.470	5.024, 246.180	4.963, 160.458	4.901, 117.627	4.840, 91.954	0.163, 162.923	
4	5.086, 503.470	5.024, 246.180	4.963, 160.458	4.901, 117.627	4.840, 91.954	5.141, 5140.881	
5	0.500, 0.500	0.500, 0.500	0.500, 0.500	0.500, 0.500	0.500, 0.500	5.141, 5140.881	
6	5.086, 503.470	5.024, 246.180	4.963, 160.458	4.901, 117.627	4.840, 91.954		
7	0.500, 0.500	0.500, 0.500	0.500, 0.500	0.500, 0.500	0.500, 0.500	0.163, 162.923	
8	0.152, 15.092	0.142, 6.899	0.129, 4.176	0.118, 2.821	0.106, 2.012		
9	0.152, 15.092	0.142, 6.899	0.129, 4.176	0.118, 2.821	0.106, 2.012	0.163, 162.923	0.152, 15.244
10	0.152, 15.092	0.142, 6.899	0.129, 4.176	0.118, 2.821	0.106, 2.012	0.163, 162.923	5.086, 508.556

[a] Beta(a, b) parameters.
[b] Equivalent data $(\tilde{x}, \tilde{n})$.

Recall that the basic event information is described by Beta(a, b) and that higher-level event information is described by the equivalent number of event occurrences $\tilde{x}$ in $\tilde{n}$ trials. Table 2 describes the 10 cases in these terms.

For each of the cases, the Bayesian analysis described in Section 4 was performed. That is, an MCMC algorithm was applied to obtain draws from the joint posterior distribution of the basic event probabilities $\mathbf{p} = (p1, p2, p3, p4, p5)$. For example, the joint posterior distributions for $\mathbf{p}$ in case 10 has the following form

$$\pi(\mathbf{p}|\mathbf{x}) \propto p1^{0.152-1}(1-p1)^{15.092-1}p2^{0.142-1}$$
$$(1-p2)^{6.899-1}p3^{0.129-1}(1-p3)^{4.176-1}p4^{0.118-1}$$
$$(1-p4)^{2.821-1}p5^{0.106-1}(1-p5)^{2.012-1}\mathrm{TE}(p)^{0.163}$$
$$(1-\mathrm{TE}(p))^{162.923-0.163}\mathrm{IE}(p)^{5.086}(1-\mathrm{IE}(p))^{508.556-5.086},$$

where TE($\mathbf{p}$) and IE($\mathbf{p}$) are functions of the basic event probabilities $\mathbf{p}$ as given in Eqs. (1) and (2), respectively. The form of the joint posterior distribution illustrates the use of basic event information represented by a beta distribution and higher-level event information represented by equivalent observational data. Once draws for $\mathbf{p}$ are obtained, TE($\mathbf{p}$) and IE($\mathbf{p}$) are evaluated resulting in draws from the posteriors of the top and intermediate event probabilities, respectively.

The results for each of the above outlined ten cases are presented in Fig. 2. For each case, the posterior distribution summaries of the TE, IE, BE2, and BE4 events are plotted; the 2.5, 50 and 97.5 percentiles are indicated by short horizontal lines. Each plot indicates the effect of including various strengths of data. For example, compare the width of the posterior 95% credible intervals for case 5 (strong TE data) versus case 7 (weak TE data) in which the stronger data have the predictable effect of reducing variability; the same holds for case 6 (strong BE1–5 data) versus case 8 (weak BE1–5 data). Note that having weak BE1–5 data is different than having very weak BE1–5 data (represented by a Beta(0.5,0.5) distribution); contrast case 2 with case 5 and case 1 with case 7. The effect of adding weak TE data can depend on the type of BE1–5 data; when there are strong BE1–5 data, there is little effect (see cases 3 and 6). However, when weak TE data is added to weak BE1–5 data, the variability of the BE2, BE4 and IE events has actually increased. Here the weak TE data do not exactly reinforce the BE1–5 data so that the resulting posterior from the combined data is wider. Also compare case 1 with case 9 and case 9 with case 10 in which wider posteriors arise when weak IE data or different strong IE data are added. Other examples of the patterns observed above can be seen in cases 1–4 in which weak and strong BE1–5 data and weak and strong TE data are considered in the four possible combinations. Thus, we see that the effect is very different depending on which level is being examined and what type of data is available at each level. This demonstrates the value of collecting different information at different levels which will be further illustrated with an application of a genetic algorithm (GA) for optimizing additional data collection based on an overall budget constraint in Section 6.

6. Optimal resource allocation

In this section, we consider the optimal allocation of additional tests performed to maximize the information gain under a fixed budget. In our example, this means that we must decide how many tests of each event should be performed in order to minimize the uncertainty in the top event probability estimate, under a fixed budget for specified costs for each event test. To achieve this optimization task, we use a GA [24,25].

Thus, we assume that there is a cost for collecting additional event data and that higher-level event data are more costly than basic event data. Consider the following costs as an example of the costs for collecting a single observations (events):

BE1: \$1
BE2: \$1

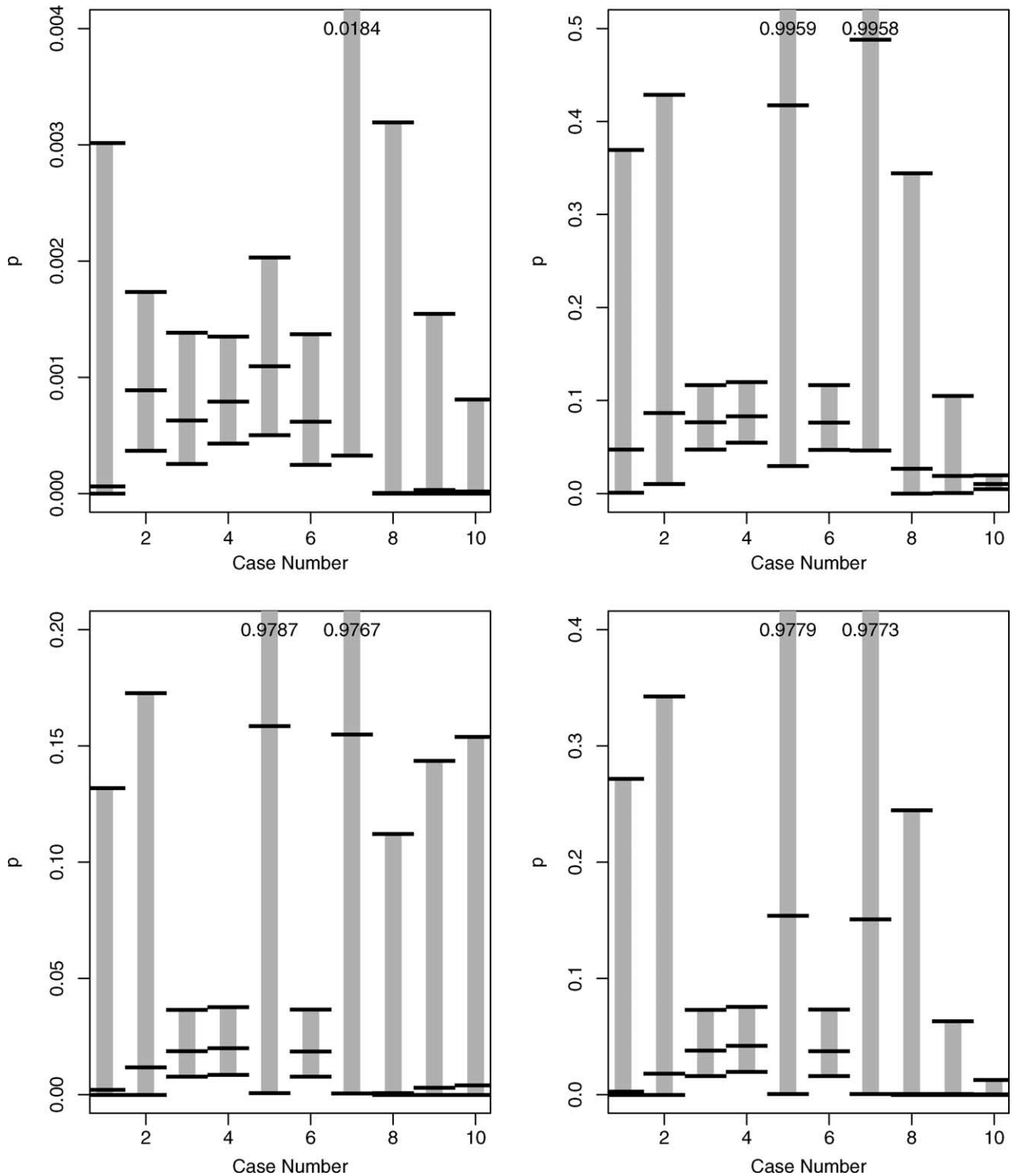

Fig. 2. Posterior 95% credible intervals for the probability of events for each of the 10 cases. The upper left panel is for the top event (TE) probability, the upper right is for the intermediate event (IE) probability, the lower left is for the basic event BE2 probability, and the lower right is for the basic event BE4 probability. The short horizontal lines correspond to the 2.5, 50 and 97.5 percentiles. Note that the very long intervals extend beyond the graphs.

BE3: $1
BE4: $1
BE5: $1
TE: $10
IE: $3

We define the maximum information gain in terms of the maximum reduction in uncertainty of the top event probability. That is, we consider the maximum reduction in the relative length of the central 90% credible interval from the top event posterior distribution before and after

taking additional data. Note that this interval itself has a distribution and we are concerned with the ratio of the 'after' new data and 'before' new data posterior lengths. Here we take the 0.75 quantile of this distribution as the criterion we wish to minimize.

Briefly, we describe how a GA can be used to find a nearly optimal allocation. A GA operates on a 'population' of candidate 'solutions' to the optimization problem. In this context, each candidate solution is a string of seven sample sizes corresponding to additional tests to be done for events BE1–BE5, TE and IE, respectively.

More specifically, the GA begins by constructing an initial population of M solutions by randomly generating solutions that do not exceed the given fixed budget. The information gain criterion for each of the solutions in the initial population is evaluated and the solutions are ranked from smallest to largest, i.e. the smallest ratio is the best solution in the initial population.

The second (and subsequent) GA generations are now populated using the two genetic operations: crossover and mutation. Crossover occurs when two parent solutions are randomly selected without replacement from the initial population according to probabilities that are inversely proportional to their rank among the M solutions. The new solution is obtained from the parent solutions by randomly picking one of the two parents and taking its sample size for the first event and then repeating this operation for each of the remaining events. The two parents are then returned to the initial population before the next crossover operation is performed. In this way, an additional M solutions are constructed using the crossover operator. Note that the solutions are checked to make sure they do not exceed the budget, so that solutions are generated until there are M such feasible solutions. The information gain criterion is also evaluated for each of these new solutions.

The GA proceeds next by mutating each of the initial M solutions; i.e. we apply genetic mutation to each of the event sample sizes. We also incorporate relaxation in the probability that mutation occurs as a function of generation.

It is desired to mutate event sample size with probability that decays exponentially as a function of generation. That is, mutations become less and less likely as the number of generations increases. To accomplish this, at generation g each event sample size is mutated with probability $\exp(-\mu g)$ where μ is a user-specified mutation rate parameter. The effect of μ is to control the rate at which mutations occur and how mutations become less and less likely as the number of generations increases. For our example, we set $\mu = 0.01$, although there is little effect on the performance of the GA by using a different value of μ [25].

Given that mutation of an event sample size occurs, the GA mutates the value with expectation approximately equal to the current event sample size and a variance that decreases with g. This is accomplished by means of a logit transformation computed through the following steps:

1. Compute $z = (y - L)/(U - L)$ where y, L, and U are the current, minimum and maximum sample sizes.
2. Compute $L = 0$, $\underline{U}$ = floor (budget/cost of event), where *floor* is the largest integer not exceeding its argument.
3. Calculate $d = \log[z/(1 - z)] + [\text{Uniform}(0, 1) - 0.5]\sigma \exp(-\mu g)$, where Uniform(0,1) denotes a random draw from a uniform distribution. Here σ is a user-specified parameter that controls the rate at which the variance decreases as a function of g.
4. Finally, compute $u = L + (U + 1 - L)\exp(d)/[1 + \exp(d)]$.

The desired mutated sample size is floor(u) which lies between L and U. The resulting logit transformation has the properties that the expected value is approximately equal to the current sample size y and the standard deviation decreases with g. Following this mutation procedure, M additional solutions satisfying the budget constraint are generated and the information gain criterion for each is evaluated.

The GA used here is 'elitist', which means that the population in the next generation consists of the M best solutions from the $3M$ solutions currently being considered (M initial solutions, M crossover solutions and M mutated solutions). We execute the above GA for G generations.

To illustrate the GA for the allocation problem described above, we consider a fixed budget of \$100. Populations of size 25 ($M = 25$) were used to generate 100 generations ($G = 100$). We consider case 8 from Section 5 in which there were no data at the intermediate and top events. The length of the 90% credible interval for top event probability based on the existing data is 0.00318. The information gain criterion is taken to be the 0.75 quantile of the relative length distribution. For our example, we chose $K = 500$, so that we want the 125th largest relative length. Thus, we take 500 draws from the joint posterior distribution of the seven event probabilities based on the current information. For each draw, the numbers of events occurring are drawn from binomial distributions using these event probabilities for the proposed sample sizes specified by the GA candidate solutions. Then the resulting posterior distribution is calculated using MCMC; we compute the 90% central credible interval for the top event probability based on 1000 draws; the length of the new interval is computed and the relative length is computed by dividing it by the length of the existing interval, 0.00318. Thus, there are 500 relative lengths, one for each of the 500 draws from the joint posterior distribution of the seven event probabilities based on the existing information.

For a budget of \$100, what resource allocation yields the most reduction in the 90% credible interval length of the top event probability? Based on a GA as described above, the GA produced the traces presented in Figs. 3 and 4 which display the criterion and additional number of tests allocated, respectively. The information gain criterion starts at 0.69 in generation 1 and decreases to 0.50 in generation 100 with an allocation of 54, 11, 5, 22 and 6 additional tests

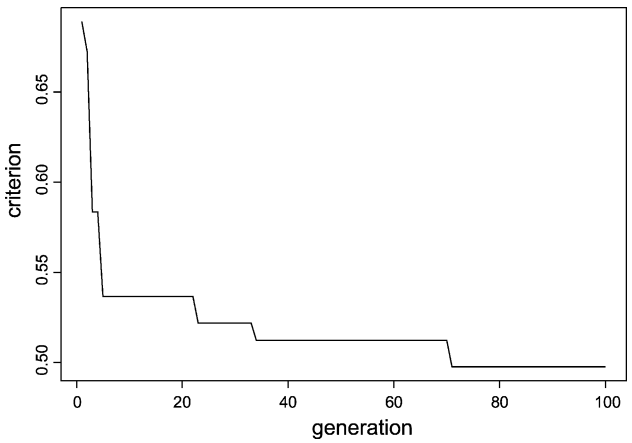

Fig. 3. GA criterion trace for first scenario.

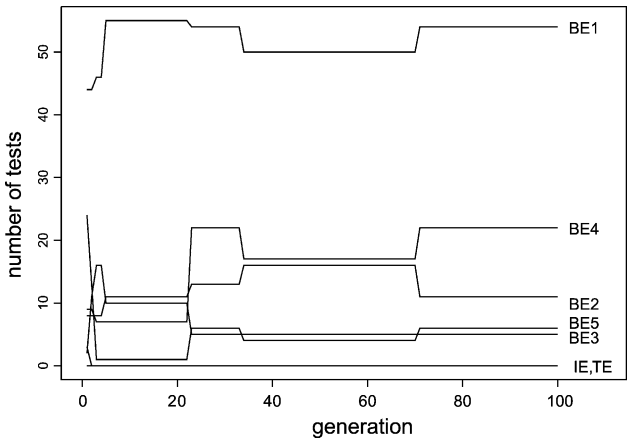

Fig. 4. GA number of tests allocation trace for first scenario.

to basic events BE1–BE5, respectively, and no allocation to either of the higher-level events. Note that an allocation using the entire budget was not identified because the slight improvement over that found was within the simulation error of the information gain criterion.

Figs. 5 and 6 provide the GA traces for the criterion and number of tests allocated for a budget of $100 but where

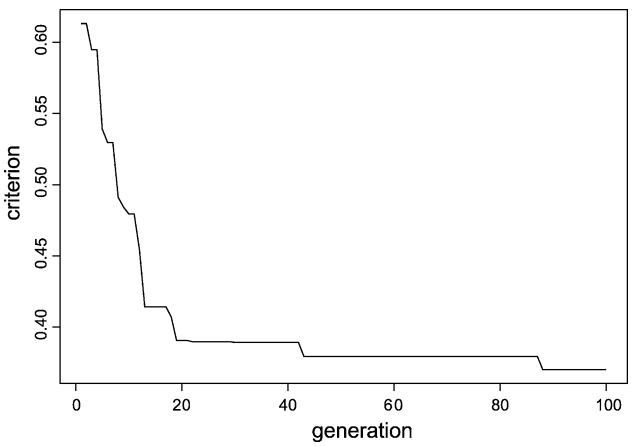

Fig. 5. GA criterion trace for second scenario.

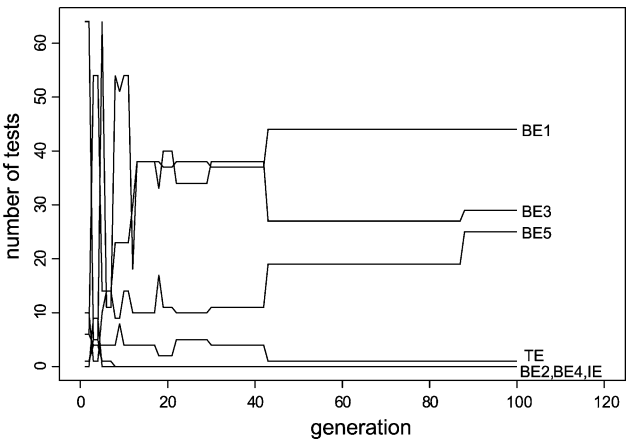

Fig. 6. GA number of tests allocation trace for second scenario.

the higher-level event costs are assumed to be $1.25 and $2 for IE and TE. The information gain criterion starts at 0.61 in generation 1 and decreases to 0.37 in generation 100 with an allocation of 44, 29 and 25 additional tests to basic events BE1, BE3 and BE5, respectively, and 1 additional test at the higher-level event TE. Consequently, no additional tests are allocated to basic events BE2 and BE4 and higher-level event IE.

7. Discussion

A fully Bayesian methodology has been developed for using multilevel event data in fault tree quantification. The method requires the identification and use of state-of-knowledge uncertainty distributions for the probabilities of occurrence of the initial basic events. The higher-level event information must be expressed as equivalent observational data. The performance of the methodology was illustrated for a simple example and it performed well. This example demonstrates the utility of the combined use of higher-level data, particularly when the initial basic event data are weak.

The methodology developed for analyzing multilevel fault tree data was then extended to address the question of how to allocate additional test resources across the fault tree events for the purpose of minimizing the uncertainty of the top event probability. That is, for a given budget, the allocation providing the most gain in information can be determined. We demonstrated how a GA provides a practical way to accomplish this.

Thus, the fully Bayesian approach is very attractive and easy to use for fault tree analysis. It can naturally handle data at different event levels. Moreover, allocation of additional resources can easily be accomplished.

Acknowledgements

We thank Dee Won for her encouragement of this work. We also thank the referee for helpful comments on an earlier version.

References

[1] Vesely WE, Goldberg FF, Roberts NH, Haasl DF. Fault tree handbook. NUREG-0492; January 1981.

[2] Hickman JW. PRA procedures guide: a guide to the performance of probabilistic risk assessments for nuclear power plants. American Nuclear Society and Institute of Electrical and Electronic Engineers, NUREG/CR-2300, vol. 1; January 1983.

[3] Russell KD, Atwood CL, Galyean WJ, Sattison MB, Rasmuson DM. Systems analysis programs for hands-on integrated reliability evaluations (SAPHIRE) version 5.0. technical reference manual, EGG-2716 (NUREG/CR-6116), July, vol. 1. Idaho National Engineering Laboratory; 1994.

[4] Russell KD, Kvarfordt KJ, Skinner NL, Wood ST, Rasmuson DM. Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) version 5.0. Integrated Reliability and Risk Analysis System (IRRAS) reference manual, EGG-2716 (NUREG/CR-6116), Idaho National Engineering Laboratory, vol. 2; July 1994.

[5] VanHorn RL, Russell KD, Skinner NL. Systems analysis programs for hands-on integrated reliability evaluations (SAPHIRE) version 5.0. Integrated Reliability and Risk Analysis System (IRRAS) tutorial manual, EGG-2716 (NUREG/CR-6116), Idaho National Engineering Laboratory, vol. 3; July 1994.

[6] Martz HF, Almond RG. Using higher-level failure data in fault tree quantification. Reliab Engng Syst Safety 1997;56:29–42.

[7] Grant GM, Roesener WS, Hall DG, Atwood CL, Gentillon CD, Wolf TR. High-pressure coolant injection (HPCI) system performance, 1987–1993. INEL-94/0158, Idaho National Engineering Laboratory; February 1995.

[8] Mastran DV. Incorporating component and system test data into the same assessment: a Bayesian approach. Op Res 1976;24: 491–9.

[9] Mastran DV, Singpurwalla ND. Incorporating component and system test data into the same assessment: a Bayesian approach. Op Res 1976; 24:491–9.

[10] Martz HF, Waller RA, Fickas ET. Bayesian reliability analysis of series systems of binomial subsystems and components. Technometrics 1988;30:143–54.

[11] Martz HF, Waller RA. Bayesian reliability analysis of complex series/parallel systems of binomial subsystems and components. Technometrics 1990;32:407–16.

[12] Johnson V, Graves T, Hamada M, Reese CS. In: Bernardo JM, Bayarri MJ, Berger JO, Dawid AP, Heckerman D, Smith AFM, West M, editors. A hierarchical model for estimating the reliability of complex systems, Bayesian statistics 7. London: Oxford University Press; 2003. p. 199–213.

[13] Simon HA, Ando A. Aggregation of variables in dynamic systems. Econometrica 1961;29:111–38.

[14] Ijiri Y. Fundamental queries in aggregation theory. J Am Stat Assoc 1971;66:766–82.

[15] Chipman JS. Optimal aggregation in large-scale econometric models. Sankhya: Indian J Stat, Ser C 1975;37:121–59.

[16] Theil H. Linear aggregation of economic relations. Amsterdam: North Holland; 1954.

[17] Mosleh A, Bier VM. On decomposition and aggregation error in estimation: some basic principles and examples. Risk Anal 1992;12: 203–14.

[18] Bier VM. On the concept of perfect aggregation in Bayesian estimation. Reliab Engng Syst Safety 1994;46:271–81.

[19] Azaiez MN, Bier VM. Aggregation error in Bayesian estimation of reliability systems. Mgmt Sci 1996;42:516–28.

[20] Gelfand AE, Smith AFM. Sampling-based approaches to calculating marginal densities. J Am Stat Assoc 1990;85:398–409.

[21] Casella G, George EI. Explaining the Gibbs sampler. Am Statistician 1992;46:167–74.

[22] Chib S, Greenberg E. Understanding the Metropolis-Hastings algorithm. Am Statistician 1995;49:327–35.

[23] Gilks WR, Richardson S, Spiegelhalter DJ. Markov chain Monte Carlo in practice. London: Chapman & Hall; 1996.

[24] Goldberg DE. Genetic algorithms in search, optimization and machine learning. New York: Addison-Wesley; 1989.

[25] Michalewicz Z. Genetic algorithms + data structures = evolution programs. New York: Springer; 1992.